

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

CLIPPEDIMAGE= JP405250377A

PAT-NO: JP405250377A

DOCUMENT-IDENTIFIER: JP 05250377 A

TITLE: SCHEDULING SYSTEM

PUBN-DATE: September 28, 1993

INVENTOR-INFORMATION:

NAME

MARUYAMA, FUMIHIRO

MINODA, YORIKO

SAWADA, HIDEHO

TAKIZAWA, YUKA

ASSIGNEE-INFORMATION:

NAME

FUJITSU LTD

COUNTRY

N/A

APPL-NO: JP04046895

APPL-DATE: March 4, 1992

INT-CL (IPC): G06F015/21

ABSTRACT:

PURPOSE: To efficiently solve a scheduling problem such as a production planning problem by utilizing failure information which is acquired up to a halfway point of processing.

CONSTITUTION: Restriction conditions regarding the startable and deadline time of a procedure and the precedence relation of the procedure and restriction conditions where plural procedures are not assigned to individual resources of a machine, etc., processing the procedure at the same time are inputted. This system is provided with an initial restriction violation condition setting part 1 which sets restriction violation conditions according to those restriction conditions, a start time change part 2 which changes the start time so that the restrictions are not violated, a start time determination part 3 which determines start time for undetermined start time without violation, and a restriction violation condition generation part 4 which generates new restriction violation conditions unless the start time can be changed or determined without violation. Sch duling is carried on according to the restriction violation conditions which are generated by the initial restriction

violation condition setting part 1 and restriction violation condition
generation part 4 and stored in the restriction violation condition storage
part 5.

COPYRIGHT: (C)1993,JPO&Japio

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平5-250377

(43)公開日 平成5年(1993)9月28日

(51)Int.Cl.³

G 0 6 F 15/21

識別記号

庁内整理番号

L 7925-5L

F I

技術表示箇所

審査請求 未請求 請求項の数 6 (全 27 頁)

(21)出願番号 特願平4-46895

(22)出願日 平成4年(1992)3月4日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 丸山 文宏

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(72)発明者 箕田 依子

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(72)発明者 澤田 秀穂

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74)代理人 弁理士 大菅 義之 (外1名)

最終頁に続く

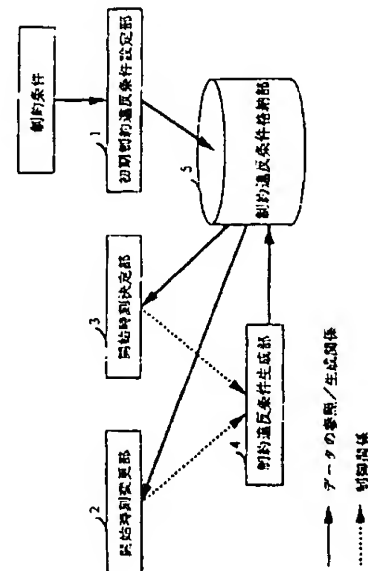
(54)【発明の名称】 スケジューリング方式

(57)【要約】 (修正有)

【目的】処理の途中時点までに得られた失敗情報を利用し、効率的に生産計画問題等のスケジューリング問題を解決する。

【構成】手順の開始可能時刻締め切り時刻、及び手順の先行関係に関する制約条件、並びに手順を処理する機械等の個々の資源に対して複数の手順が同時に割り当てられないという制約条件を入力する。これら制約条件から制約違反条件を設定する初期制約違反条件設定部1、違反しないように開始時刻を変更する開始時刻変更部2、違反しないように未定の開始時刻に対し開始時刻を決定する開始時刻決定部3、さらに違反しないように開始時刻を変更又は決定できない場合に、新たな制約違反条件を生成する制約違反条件生成部4を設ける。初期制約違反条件設定部1及び制約違反条件生成部4で生成した制約違反条件が格納された制約違反条件格納部5の条件に従って、スケジュールを行う。

本発明の原理構成図(その1)



【特許請求の範囲】

【請求項1】 それぞれ所要時間を持ち、開始可能時刻に関する制約条件、締め切り時刻に関する制約条件、および、先行関係に関する制約条件が与えられた複数の手順について、これらの制約条件、および手順を処理する機械等の個々の資源に対して複数の手順が同時に割り当てられないことがないという制約条件を満たすように、それぞれの開始時刻を決定するスケジューリング方式において、

該制約条件から制約違反の十分条件となる制約違反条件を最初に設定する初期制約違反条件設定部1と、既に開始時刻が決まっている手順の開始時刻を変更する開始時刻変更部2と、

開始時刻が未定の手順についてその開始時刻を決定する開始時刻決定部3と、

前記開始時刻変更部2および前記開始時刻決定部3において、当該手順の開始時刻をどのように設定しても少なくとも1つの制約違反条件が成立する場合に、それぞれの開始時刻に関して成立する制約違反条件中の当該手順の開始時刻に対応する変数に当該開始時刻を代入した条件の論理積として新しい制約違反条件を生成する制約違反条件生成部4と、

前記初期制約違反条件設定部1および前記制約違反条件生成部4が生成した制約違反条件を蓄積する制約違反条件格納部5と、

を備え、手順の開始時刻を変更／決定する際には前記制約違反条件格納部5に格納されている制約違反条件をどれも成立させないように行なうことを特徴とするスケジューリング方式。

【請求項2】 請求項1記載のスケジューリング方式であって、制約違反条件中に現れる各手順の開始可能時刻および締め切り時刻を変数とすることにより、開始可能時刻や締め切り時刻の変更があった場合にも変更前に生成した制約違反条件を有効とするスケジューリング方式。

【請求項3】 請求項1および請求項2記載のスケジューリング方式であって、一つのスケジュールが得られる度に開始可能時刻を示す変数または締め切り時刻を示す変数の値をより厳しい値に更新し、最終的にスケジュール不能となって開始可能時刻を示す変数または締め切り時刻を示す変数の他には変数を含まない制約違反条件を生成することにより、最短総所要時間スケジュール等の最適スケジュールを求めるスケジューリング方式。

【請求項4】 それぞれ所要時間を持ち、開始可能時刻に関する制約条件、締め切り時刻に関する制約条件、および、先行関係に関する制約条件が与えられた複数の手順について、これらの制約条件、および手順を処理する機械等の個々の資源に対して複数の手順が同時に割り当てられないことがないという制約条件を満たすように、それぞれの開始時刻を決定するスケジューリング方式に

において、

該制約条件から制約充足の必要条件となる制約充足条件を最初に設定する初期制約充足条件設定部6と、

既に開始時刻が決まっている手順の開始時刻を変更する開始時刻変更部7と、

開始時刻が未定の手順についてその開始時刻を決定する開始時刻決定部8と、

前記開始時刻変更部7および前記開始時刻決定部8において、当該手順の開始時刻をどのように設定しても少なくとも1つの制約充足条件が成立しない場合に、それぞれの開始時刻に関して成立しない制約充足条件中の当該手順の開始時刻に対応する変数に当該開始時刻を代入した条件の論理和として新しい制約充足条件を生成する制約充足条件生成部9と、

前記初期制約充足条件設定部6および前記制約充足条件生成部9が生成した制約充足条件を蓄積する制約違反条件格納部10と、

を備え、手順の開始時刻を変更／決定する際には前記制約充足条件格納部10に格納されている制約充足条件をすべて満たすように行なうことを特徴とするスケジューリング方式。

【請求項5】 請求項4記載のスケジューリング方式であって、制約充足条件中に現れる各手順の開始可能時刻および締め切り時刻を変数とすることにより、開始可能時刻や締め切り時刻の変更があった場合にも変更前に生成した制約充足条件を有効とするスケジューリング方式。

【請求項6】 請求項4および請求項5記載のスケジューリング方式であって、一つのスケジュールが得られる度に開始可能時刻を示す変数または締め切り時刻を示す変数の値をより厳しい値に更新し、最終的にスケジュール不能となって開始可能時刻を示す変数または締め切り時刻を示す変数の他には変数を含まない制約充足条件を生成することにより、最短総所要時間スケジュール等の最適スケジュールを求めるスケジューリング方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、生産計画問題等のスケジューリング問題に関する。更に詳しくは、スケジューリング処理中に生成した制約違反条件または制約充足条件をその後の処理でも有効に利用し、効率よくスケジュールを作成することを可能とするスケジューリング方式に関する。

【0002】

【従来の技術】生産計画等のスケジューリング問題には、例えば、ジョブジョップスケジューリング問題がある。

【0003】ジョブジョップスケジューリング問題では、 n 個の仕事がそれぞれ m 個の手順で構成されており、各手順は m 台の機械のいずれか一つによって処理さ

3

れる。各機械は同時に二つ以上の手順を実行することはできず、各手順には、所要時間、開始可能時刻、締め切り時刻がある。また、一つの仕事を構成する手順の間には先行関係がある。このとき、総所要時間が最も短くなるように手順の開始時刻を決定することが問題である。

【0004】このようなスケジューリング問題は、潜在的な解空間が組合せ的に大きくなり、単純な探索方式は適用できない。ところが、見かけ上の解空間のなかで求める解が存在しない部分が識別できるケースがあり、このような構造を有効に利用することにより効率的に解を求めることが可能になる場合がある。

【0005】従来の技術としては、(a) 整数計画法、(b) 分枝限定法、(c) ATMSがある。ジョブスケジューリング問題において総所要時間を最短にする例で説明する。

(a) 整数計画法

各仕事の各手順の開始時刻または終了時刻に対応する変数に加えて、二つの手順が同時に同一機械で処理されることがないという条件を表現するために、同一機械で実行される手順の間の前後関係に対応する0-1変数を導入する。ところが、この処理により変数の数が増え、整数計画法の計算時間は変数の数について指数的に増加する傾向があるため、計算時間が大幅に延びてしまう。

(b) 分枝限定法

一部の手順だけの開始時刻が決定されている状態（部分スケジュール）において、まだスケジュールされていない手順の処理時間の総和（個々の仕事に注目した総和および個々の機械に注目した総和）から総所要時間の下界値を求め、既に求まっている完全スケジューリングの総所要時間より長い場合には、その部分スケジュールを含むスケジュールは探索対象から除く。ところが、この方法では、総所要時間の下界値を見積もるのに、個々の仕事に注目するか個々の機械に注目してまだスケジュールされていない手順の所要時間を計算するため、複数の仕事の絡みを考慮に入れることができない。従って、下界値が実際より小さく見積もられることになり、探索範囲の絞り込みが効率的に行えない。

(c) ATMS

同一の機械で実行される手順の間の前後関係を仮とし、制約条件を満たすことができなかった仮説の組合せを格納しておき、それを含むような組合せは捨てる。ところが、仮説の組合せの数が増大するため、扱える規模が限られる。

【0006】

【発明が解決しようとする課題】しかしながら、以上の3種類の従来方式には、前述したようにそれぞれ問題があった。

【0007】すなわち、整数計画法では、変数の数が増えるために計算時間が指数的に増加するという問題があり、また、分枝限定法では、下界値の見積もりが問題と

4

なる。さらに、ATMSでは、仮説の組合せ数の増加による計算時間の増大が問題である。

【0008】本発明は、開始可能時刻に関する制約条件、締め切り時刻に関する制約条件、および、先行関係に関する制約条件を与えられた複数の手順について、これらの制約条件、および、手順を処理する機械等の個々の資源に対して複数の手順が同時に割り当てられないことがないという制約条件を満たすように、それぞれの開始時刻を決定するスケジューリング問題を対象とし、処理の途中時点までに得られた失敗情報をそれ以降の処理で最大限に利用し、効率的にスケジューリング問題を解決することを目的とする。

【0009】

【課題を解決するための手段】本発明（請求項1）の原理構成図を図1に示す。本発明は、開始可能時刻に関する制約条件、締め切り時刻に関する制約条件、および、先行関係に関する制約条件を与えられた複数の手順について、これらの制約条件、および、手順を処理する機械等の個々の資源に対して複数の手順が同時に割り当てられないことがないという制約条件を満たすように、それぞれの開始時刻を決定するスケジューリング問題を対象とすることを前提とする。

【0010】まず、初期制約違反条件設定部1は、制約条件から制約違反の十分条件となる初期制約違反条件を設定する。設定した初期制約違反条件は、後述する制約違反条件格納部5に格納する。

【0011】次に、開始時刻変更部2は、既に開始時刻が決まっている手順の開始時刻を変更する。このとき、制約違反条件格納部5に格納されている制約違反条件をどれも成立させないように変更する。開始時刻が未定の手順に対応する値が未定の変数を含む制約違反条件は不成立と見なす。

【0012】開始時刻決定部3は、開始時刻が未定の手順についてその開始時刻を決定する。このとき、制約違反条件格納部5に格納されている制約違反条件をどれも成立させないように決定する。開始時刻が未定の手順に対応する値が未定の変数を含む制約違反条件は不成立と見なす。

【0013】制約違反条件生成部4は、前記開始時刻変更部2および前記開始時刻決定部3において制約違反条件が成立してしまう場合に、新しい制約違反条件を生成する。生成した制約違反条件は後述する制約違反条件格納部5に格納する。

【0014】制約違反条件格納部5は、処理の開始直後には前記初期制約違反条件設定部1が設定した初期制約違反条件のみを格納し、その後、前記制約違反条件生成部4が生成した制約違反条件を順次格納する。

【0015】本発明（請求項4）の原理構成図を図2に示す。まず、初期制約充足条件設定部6は、制約条件から制約充足の必要条件となる初期制約充足条件を設定す

る。設定した初期制約充足条件は、後述する制約充足条件格納部10に格納する。

【0016】次に、開始時刻変更部7は、既に開始時刻が決まっている手順の開始時刻を変更する。このとき、制約充足条件格納部10に格納されている制約充足条件をすべて満たすように変更する。開始時刻が未定の手順に対応する値が未定の変数を含む制約充足条件は成立していると見なす。

【0017】開始時刻決定部8は、開始時刻が未定の手順についてその開始時刻を決定する。このとき、制約充足条件格納部5に格納されている制約充足条件をすべて満たすように決定する。開始時刻が未定の手順に対応する値が未定の変数を含む制約充足条件は成立していると見なす。

【0018】制約充足条件生成部9は、前記開始時刻変更部7および前記開始時刻決定部8において制約充足条件が不成立の場合に、新しい制約充足条件を生成する。生成した制約充足条件は後述する制約充足条件格納部10に格納する。

【0019】制約充足条件格納部10は、処理の開始直後には前記初期制約充足条件設定部6が設定した初期制約充足条件のみを格納し、その後、前記制約充足条件生成部10が生成した制約充足条件を順次格納する。

【0020】

【作用】次に、図1に示した請求項1の原理構成の作用を説明する。本原理構成では、開始可能時刻に関する制約条件、締め切り時刻に関する制約条件、および、先行関係に関する制約条件を与えられた複数の手順について、これらの制約条件、および、手順を処理する機械等の個々の資源に対して複数の手順が同時に割り当てられないという制約条件を満たすように、それぞれの開始時刻を決定するスケジューリング問題を対象とする。

【0021】まず、初期制約違反条件設定部1は、与えられた制約条件の否定を取ることで、初期の制約違反条件を設定する。そして、設定した初期制約違反条件は制約違反条件格納部5に格納する。

【0022】次に、制約違反条件格納部5に格納されている制約違反条件を探索し、成立している制約違反条件がある場合には開始時刻変更部2を起動する。開始時刻変更部2は、制約違反条件を成立させないように開始時刻を変更する。ここで、制約違反条件を成立させないような開始時刻が存在しないときには制約違反条件生成部4を起動する。制約違反条件生成部4は、成立してしまう制約違反条件の論理積をとることにより新しい制約違反条件を生成し、制約違反条件格納部5に格納する。

【0023】一方、開始時刻変更部2の処理において制約違反条件を成立させないように開始時刻を変更でき、しかも開始時刻が未定の手順がない場合にはそのスケジューリングは成功である。これによって条件を満たすス

ケジュールが生成されたことになる。

【0024】まだ開始時刻が未定の手順が残っている場合には開始時刻決定部3を起動する。開始時刻決定部3は制約違反条件を成立させないように開始時刻を決定する。ここで制約違反条件を成立させないように開始時刻を決定でき、しかも開始時刻が未定の手順が残っていない場合にはそのスケジューリングは成功である。制約違反条件を成立させないように開始時刻を決定できない場合には制約違反条件生成部4を起動する。制約違反条件生成部4は、前述と同様に、成立してしまう制約違反条件の論理積をとることにより新しい制約違反条件を生成し、制約違反条件格納部5に格納する。

【0025】制約違反条件格納部5に新たな制約違反条件を格納した後、再び、開始時刻変更部2および開始時刻決定部3を起動するが、このとき、変更あるいは決定するような開始時刻の変数がのこっていない場合には、スケジューリングは失敗したことになる。

【0026】以上の作用のように、生成した制約違反条件は、必ず制約違反が起きてしまう、制約違反の十分条件となる。一つの手順について開始時刻をどのように設定しても少なくとも一つの制約違反条件が成立してしまうとき、当該手順に対応する変数が数値化された制約違反条件を生成することにより、開始時刻を変更すべき手順を絞り込むとともに、その開始時刻の範囲を絞り込む。また、複数の不等式の論理積を取ることで、複数の仕事の絡みにより生じる制約違反のための十分条件が表現できる。

【0027】次に、図2に示した請求項4の原理構成の作用を説明する。まず、初期制約充足条件設定部6は、与えられた制約条件から制約充足の必要条件となる初期の制約充足条件を設定する。そして、設定した初期制約充足条件は制約充足条件格納部10に格納する。

【0028】次に、制約充足条件格納部10に格納されている制約充足条件を探索し、成立していない制約充足条件がある場合には開始時刻変更部7を起動する。開始時刻変更部7は、制約充足条件をすべて満たすように開始時刻を変更する。ここで、制約充足条件を満たすような開始時刻が存在しないときには制約充足条件生成部9を起動する。制約充足条件生成部9は、成立しない制約充足条件の論理和をとることにより新しい制約充足条件を生成し、制約充足条件格納部10に格納する。

【0029】一方、開始時刻変更部7の処理において制約充足条件をすべて満たすように開始時刻を変更でき、しかも開始時刻が未定の手順がない場合にはそのスケジューリングは成功である。これによって条件を満たすスケジュールが生成されたことになる。

【0030】まだ開始時刻が未定の手順が残っている場合には開始時刻決定部8を起動する。開始時刻決定部8は制約充足条件をすべて満たすように開始時刻を決定する。ここで制約充足条件を満たすように開始時刻を決定

でき、しかも開始時刻が未定の手順が残っていない場合にはそのスケジューリングは成功である。制約充足条件を満たすように開始時刻を決定できない場合には制約充足条件生成部9を起動する。制約充足条件生成部9は、前述と同様に、成立してしまう制約充足条件の論理和をとることにより新しい制約充足条件を生成し、制約充足条件格納部10に格納する。

【0031】制約充足条件格納部10に新たな制約充足条件を格納した後、再び、開始時刻変更部7および開始時刻決定部8を起動するが、このとき、変更あるいは決定するような開始時刻の変数が残っていない場合には、スケジューリングは失敗したことになる。

【0032】以上の作用のように、生成した制約充足条件は、制約条件を満たすスケジュールが存在するためには満たされなくてはならない制約充足の必要条件となる。一つの手順について開始時刻をどのように設定しても少なくとも一つの制約充足条件が成立しないとき、当該手順に対応する変数が数値化された制約充足条件を生成することにより、開始時刻を変更すべき手順を絞り込むとともに、その開始時刻の範囲を絞り込む。また、複数の不等式の論理和を取ることで、複数の仕事の絡みにより生じる制約違反が起きないための必要条件が表現できる。

【0033】図1に示した請求項1の原理構成において、制約違反条件中に現れる各手順の開始可能時刻および締め切り時刻を変数として同様の作用を実行することにより、開始可能時刻または締め切り時刻の変更が起きた場合でも、値が変更された変数の値を代えるだけで変更以前に生成した制約違反条件をそのまま適用することができる（請求項2）。

【0034】図1に示した請求項1の原理構成の作用において、一つのスケジュールが得られる度に開始可能時刻を示す変数または締め切り時刻を示す変数の値をより厳しい値に更新し、スケジューリング処理を繰り返す。最終的にスケジュール不能となって開始可能時刻を示す変数または締め切り時刻を示す変数の他には変数を含まない制約違反条件を生成することにより、必ず制約違反が起きるような開始可能時刻や締め切り時刻の関係が表現され、最短総所要時間スケジュール等の最適スケジュールが求まる（請求項3）。

【0035】図2に示した請求項4の原理構成において、制約充足条件中に現れる各手順の開始可能時刻および締め切り時刻を変数として同様の作用を実行することにより、開始可能時刻または締め切り時刻の変更が起きた場合でも、値が変更された変数の値を代えるだけで変更以前に生成した制約充足条件をそのまま適用することができる（請求項5）。

【0036】図2に示した請求項4の原理構成の作用において、一つのスケジュールが得られる度に開始可能時刻を示す変数または締め切り時刻を示す変数の値をより

厳しい値に更新し、スケジューリング処理を繰り返す。最終的にスケジュール不能となって開始可能時刻を示す変数または締め切り時刻を示す変数の他には変数を含まない制約充足条件を生成することにより、制約充足を満たすスケジューリングが可能のための開始可能時刻や締め切り時刻の関係が表現され、最短総所要時間スケジュール等の最適スケジュールが求まる（請求項6）。

【0037】

【実施例】図3は、本発明の一実施例のシステム構成図である。本実施例は、図1に示した原理構成の実施例を示している。CPU310、I/Oインタフェース320、主記憶330、入出力装置340、記憶装置350等を備えた計算機システム300で構成することが可能である。図2に示した原理構成の場合でも同様のシステム構成で実現できる。

【0038】主記憶330内には、原理構成で示した各部、すなわち、初期制約違反条件設定部331、変数値変更部332、変数値決定部333、制約違反条件生成部334に対応するソフトウェアを置き、そのソフトウェアに従ってCPU310によりスケジューリング処理を実行する。記憶装置350内には制約違反条件格納部351を置き、初期制約違反条件設定部331や制約違反条件生成部334で生成した制約違反条件を格納する。制約違反条件格納部351は主記憶330内に置くことも可能である。次に、このシステムの動作を図4の動作フローチャートに沿って説明する。

【0039】ジョブショップスケジューリング問題を解こうとするユーザは、まず、問題となる制約条件を入出力装置340から入力する。ここで、主記憶330内の初期制約違反条件設定部331が起動され、初期制約違反条件設定部331は入力された全制約条件から制約違反条件を生成し、制約違反条件格納部351に格納する(S1)。

【0040】次に、初期スケジュールを生成する(S2)。初期スケジュールは、入力された制約条件を一応満たすスケジュールであり、最適なスケジュールか否かは不明である。この初期スケジュールは、自明のスケジューリング手法、例えばGiffler-Thompsonの手法等で求めることができる（『整数計画法と組合せ最適化』、日科技連合、1982年、291ページ）。

【0041】本実施例では、この初期スケジュールと初期制約違反条件をもとにスケジューリングを実行する。まず、制約違反条件格納部351のなかの制約違反条件を探索し、成立している制約違反条件NJがあるか否かを判定する(S3)。成立しているNJが存在する場合(Y)には、変数値変更部332を起動する。変数値変更部332では、すべての制約違反条件が成立しないように開始時刻を変更する(S4)。そして、この変更処理において制約違反条件が成立しないような開始時刻が存在したか否かを次に判定する(S5)。存在しない、すなわち、どのように開始時刻を変更してもいずれかの制約違反条件を満たしてし

まう場合（開始時刻がすべて禁止(Y)）の場合には制約違反条件生成部334を起動する(S9～S10)。この処理については後述する。

【0042】一方、開始時刻変更処理(S4)において制約違反条件が成立しないような開始時刻が存在した場合(S5のN)には、次に、開始時刻未定の手順があるか否かを判定する(S6)。未定の手順がない場合(N)には、すべての手順の開始時刻が制約違反条件が成立しないように決定できたことになり、スケジューリング成功として処理を終了する。

【0043】一方、未定の手順がある場合(Y)には変数値決定部333を起動する。変数値決定部333は、未定の開始時刻を、制約違反条件が成立しないように決定する(S7)。そして、そのような開始時刻が決定できたか否かをS5に戻り判定する。

【0044】開始時刻変更処理S4および開始時刻決定処理S5において制約違反条件を成立させないような変数値設定ができない場合(S5のY)には、制約違反条件生成部334を起動する。

【0045】制約違反条件生成部334は開始時刻変更処理S4あるいは開始時刻決定処理S5において成立してしまう制約違反条件について論理積をとり、新たな制約違反条件を生成する(S8)。そして、生成した制約違反条件の左辺に変数が残っているか否かを次に判定する(S9)。残っていない場合(N)には、もうこれ以上スケジューリング処理は続けられず、スケジューリング失敗となる。一方、変数が残っている場合(Y)には生成した制約違反条件を制約違反条件格納部351に格納し、S4に戻り、スケジューリング処理を繰り返す。

【0046】以上の処理により、スケジューリング処理が実行され、最適なスケジューリングが行われる。以下、実際のジョブショップスケジューリング問題を対象に本実施例のスケジューリング方式を説明する。

【0047】問題として、3仕事2機械のジョブショップスケジューリング問題を取り上げる。この問題は、本実施例の処理の流れを説明するための小規模な問題例である。図5～図7の〔問題1〕に問題の内容を示す。

【0048】3種類の仕事のそれぞれが2つの機械によって処理を行なう場合の最適スケジュール(総所要時間最小)を求める。まず、仕事1は1番目の手順として機械1を使用し(仕事1の手順1が機械1という意味で111とする)、2番目の手順として機械2を使用する(122)。仕事1の手順1の処理時間は6、手順2の処理時間は4である。

【0049】また、仕事2は1番目の手順として機械2を使用し(212)、2番目の手順として機械1を使用し(221)、手順の処理時間は、それぞれ、手順1が4、手順2が3である。さらに、仕事3の1番目の手順は機械1(311)、2番目の手順は機械2(322)で行ない、処理時間はそれぞれ、手順1が2、手順2が

5である。

【0050】さらに、一つの機械をでは同時に2つ以上の手順は実行できない。このような制約条件をもとに各手順の開始時間を最適スケジューリングする。以下では仕事iの手順j(機械k)の開始時刻の変数を x_{ijk} で表す。

【0051】まず、初期制約違反条件設定S1を実行する。すなわち、図5の〔問題1〕に示した表のような条件から初期の制約違反条件を生成する。この問題に関する初期制約違反条件は図5の〔1〕に示すものである。初期制約違反条件には、開始可能時刻に関するもの(同図〔1〕(a))、締め切り時刻に関するもの(同図〔1〕(b))、手順の後先の制約を表す先行関係に関するもの(同図〔1〕(c))、複数の手順を同一機械で同時処理できないことについてのもの(同図〔1〕(d))がある。

【0052】開始可能時刻に関しては、例えば、111や212、311の処理は時刻0以降に開始可能なので、この条件の否定を示す $x_{111} < 0$ 、 $x_{212} < 0$ 、 $x_{311} < 0$ が制約違反条件になる。また、各仕事の2手順目は1手順目の処理が完了しないと開始できないので、122に関しては1手順目の処理時間6以降に開始可能となる。そこで、この条件の否定を示す $x_{122} < 6$ が制約違反条件となる。同様に、各仕事の各処理開始可能時間について制約違反条件が設定される(同図〔1〕(a))。

【0053】次に、締め切り時刻に関しては、締め切り時刻を示す変数Eを導入する。そして、仕事1については手順1と手順2が終了した後に締め切り時間Eがくるはずであるから、手順1と手順2の処理時間6と4を足した時刻10以降に締め切り時刻がくるはずである。よって、制約違反条件としては否定をとり $x_{111} + 10 > E$ を生成する。同様に、各仕事の各手順について締め切り時刻に関する初期制約違反条件を設定することが可能である。(同図〔1〕(b))。

【0054】また、先行関係に関しては、仕事1の手順2は手順1が完了しないと開始できないから、仕事1の手順2は、手順1の開始時刻 x_{111} に処理時間6を加えた時刻以降に開始可能である。この条件の否定をとり $x_{111} + 6 > x_{122}$ という制約違反条件が生成される。同様に、各仕事の手順の先行関係について制約違反条件が設定できる(同図〔1〕(c))。

【0055】最後に、複数の手順の同時処理不可に関しては、同一機械で複数の手順を同時に実行できないことを制約条件として示す。すなわち、111を実行している間(時刻をとり、 $x_{111} < t < x_{111} + 6$)には同じ機械で処理する手順221は実行できないことになり、 $x_{111} + 6 \leq x_{221}$ (D)が制約条件となる。一方、手順221の実行中に111を実行することもできないことから、これを表す制約条件として $x_{221} + 3 \leq$

11

x_{111} (②) という制約条件がなりたつ。①V② (①あるいは②) という制約条件から制約違反条件を生成すると、 $x_{111} + 6 > x_{211} \wedge x_{221} + 3 > x_{111}$ となる。同様にして、同一の機械で行なう全ての手順について (機械1の111と311、機械1の311と221、機械2の122と212、機械2の122と322、機械2の212と322)、同時実行不可に関する初期制約違反条件が生成できる (図4〔1〕(d))。

【0056】以上の初期制約違反条件を設定後、スケジューリング処理を開始する。スケジューリングは初期スケジュール (図8①) をもとに実行する (図4のS2)。初期スケジュールは、先に設定した制約違反条件を一応満たすように作成されたスケジュールである。この初期スケジュールによると、締め切り時刻Eは15となる。

【0057】そこで、スケジューリング処理ではE=14と、締め切り時刻Eを早めて処理を実行する。まず、初期スケジュール (図8①) をもとに、成立している制約違反条件があるか否かを初期制約違反条件を探索することにより判定する (S3)。例えば、締め切り時刻に関する初期制約違反条件 $x_{111} + 10 > E$ については、 $x_{111} = 0$ 、E=14なので制約違反条件を満たさない。このようにして成立している制約違反条件を探していくと、 $x_{322} + 5 > E$ の制約違反条件が成立していることが分かる ($x_{322} = 10$ 、E=14であるから上記条件は成立する) (図6〔2〕(1))。

【0058】そこで、次に、開始時刻変更処理S4を実行する。すなわち、成立した制約違反条件に含まれる手順の開始時刻を順に変更してみる。すると、 $x_{322} = 0 \sim 1$ では開始可能時刻についての制約違反条件 $x_{322} < 2$ を満たし、 $x_{322} = 2 \sim 7$ では $x_{311} + 2 > x_{322}$ を満たし ($x_{311} = 6$ だから)、 $x_{322} = 8 \sim 9$ では $x_{122} + 4 > x_{322} \wedge x_{322} + 5 > x_{122}$ を満たし ($x_{122} = 6$ だから)、 $x_{322} = 10$ では $x_{322} + 5 > E$ を満たすことが分かる (図6〔2〕(1))。

【0059】このことは図4のS5の処理ですべての開始時刻が禁止されたことを意味し (Y)、次に、制約違反条件を新たに生成する処理を実行する (S8)。すなわち、成立してしまった制約違反条件の x_{322} に各開始時刻を代入し、その論理積を求める。例えば、 $x_{311} + 2 > x_{322}$ については、 $x_{322} = 2 \sim 7$ の数値をそれぞれ代入した式 ($x_{311} + 2 > 2$ 、 $x_{311} + 2 > 3$ 、 \dots 、 $x_{311} + 2 > 7$) の論理積として $x_{311} > 5$ が得られる。同様の処理を成立した残りの各式 ($x_{122} + 4 > x_{322} \wedge x_{322} + 5 > x_{122}$ 、 $x_{322} + 5 > E$) でも実行し、それぞれ、 $x_{122} > 5 \wedge x_{122} < 13$ 、 $15 > E$ という論理積を得る。次に、これらの論理積の式の論理積として新たな制約違反条件が生成される (図6〔2〕(1) 中の下線付きの式)。

【0060】この新たな制約違反条件の左辺は変数を含

12

むので (S9のY)、該制約違反条件を制約違反条件格納部351に格納する (S10)。以上の処理によって、初期制約違反条件に新たに一つ制約違反条件が加わったことになる。

【0061】以下、以上の処理 (S4～S10) を繰り返すことになる。S4において開始時刻を変更する変数としてどれを選ぶかには様々な方法があるが、ここでは、最後の制約違反条件の最後の変数について変更することにする。すなわち、次には、変数 x_{122} を変更する (図6〔2〕(2))。この処理において、開始時刻をどのように変更しても何らかの制約違反条件が成立するので (S5のY)、新たな制約違反条件を生成する (S8)。この結果、図6〔2〕(2) 中の下線付きの制約違反条件が生成される。

【0062】次には、 x_{311} について変数値を変更する (図6〔2〕(3))。この場合も開始時間 x_{311} をどのような値に取っても何らかの制約違反条件が成立するので (S5のY)、新たな制約違反条件を生成する (S8)。そして、図6の〔2〕(3) 中の下線付きの制約違反条件が生成される。

【0063】次には、 x_{111} について変数値変更を試みる。すると、 $x_{111} = 2$ としたとき、制約違反条件をどれも成立させない (S5のN) (図6〔2〕(4))。S6では、開始時刻未定の手順がある (Y) ので、開始時刻決定処理 (S7) を実行する。

【0064】開始時刻決定処理 (S7) では、まず、図6の〔2〕(2) で生成された制約違反条件中にある x_{311} について決定する。全制約違反条件を成立させないように変数を決定すると、 $x_{311} = 0$ となる (図6〔2〕(5))。さらに、同様に、 x_{122} の値を決定し、8とする (図6〔2〕(6))。

【0065】次に、 x_{322} の値を決定する。すると、全制約違反条件を成立させないように x_{322} を決定することができない (S5のN)。そこで、新たに制約違反条件を生成する (S8)。すると、図6〔2〕(7) 中の下線付きの制約違反条件が得られる。そして、この制約違反条件を制約違反条件格納部351に格納した後、開始時刻変更処理S4に戻る。

【0066】まず、 x_{122} の値を変更する (S4)。すると、 $x_{122} = 9$ あるいは10のとき全ての制約違反条件を成立させない。そこで $x_{122} = 9$ とし (図7〔2〕(8))、未定の開始時刻を決定する (S7)。 $x_{221} = 8$ 、 $x_{122} = 0$ 、 $x_{322} = 4$ が決まる。以上ですべての手順の開始時刻が決定したので (S6のN)、スケジューリングは成功したことになる (図7〔2〕(9))。スケジュールは図8の②に示すようになる。

このとき、E=13である。最適スケジュールを求めるためには、次に、E=12としてS3からの処理を行なう。この処理では、まず、 $x_{122} + 4 > E$ となるので、 x_{122} について開始時刻変更を行ない (S4)、制約違

13

反条件を成立しないように変数変更ができず、S8で新たな制約違反条件を生成し(図7〔2〕(10)の下線付き制約違反条件)、制約違反条件格納部351に格納する。そして、また x_{111} の開始時刻変更処理を行なう。この場合も制約違反条件を成立しないように変数変更ができず、S8で新たな制約違反条件を生成し(図7〔2〕(11)の下線付き制約違反条件)、制約違反条件格納部351に格納する。そして、また x_{212} について開始時刻変更を行なう(S4)。この場合も制約違反条件を成立しないように変数変更ができず、S8で新たな

制約違反条件を生成する(図7〔2〕(12)の下線付き制約違反条件)。
【0067】すると、この制約違反条件の左辺に変数がなく(S9のN)、スケジューリングは失敗となる。従って、先に得られたスケジュール(図8の㊲)が最適スケジュールとなる。この結果、総所要時間は13である。

【0068】以上のように、すべての制約違反条件を満たさないように変数値を設定していき、設定できない場合には新たな制約違反条件を生成し、これに加えて変数値の設定処理を繰り返すことにより、制約条件を満たすスケジュールを得ることができる。また、一端求めたスケジュールについて、さらに締め切り時刻の制約を厳しくして処理を繰り返すことにより、最適スケジュールを得ることができる。

【0069】以上に説明した問題(問題1)は、説明を簡単化するために仕事、手順、ともに最小限に小さなものにした。以下に、少し規模の大きな問題について説明する。図9～図13にこの問題(問題2)についての処理を説明する。この問題は、『整数計画法と組合せ最適化』(日科技連、1982年)の289ページのジョブショップスケジューリング問題と同一のものである。そして、本実施例では、本発明の請求項1、請求項2、請求項3のスケジューリング方式によるもので、制約違反条件を使用して最適スケジュールを求める。以下では、変数 x_{ijk} は「仕事iの手順j(機械kで処理する)の開始時刻を表し、 l_i は手順iの所要時間を表す。

【0070】問題は4仕事、3機械による3手順の仕事の総所要時間を最小にする最適スケジュールを求めるものである。図9の〔問題2〕に示すように、各仕事の各手順に使用する機械と、その処理時間が条件として与えられている。前述の問題と同様に、同一機械で同時に複数の手順を実行することはできない。

【0071】最初に、以上の条件から初期制約違反条件を求め(S1)、制約違反条件格納部351に格納する(図9〔1〕)。そして、次に、上記の条件を満たすような初期スケジュールを求め(S2)、これをもとにスケジューリング処理を開始する(図14の初期スケジュール)。初期スケジュールでは締め切り時間 $E=43$ な

14

ので、それよりも厳しい $E=42$ として処理を開始する。

【0072】まず、成立してしまう制約違反条件を探索し(S3)、その制約違反条件に含まれている変数について開始時刻変更処理(S4)を実行する(図10〔2〕(1))。しかし、すべての制約違反条件を成立させないような変数値は存在しないので、制約違反条件生成処理(S8)を実行し、制約違反条件格納部351に格納する(図10〔2〕(1)中の下線付き制約違反条件)。

【0073】次に、開始時刻が未定の変数について変数値決定処理(S7)を行なう(図10〔2〕(2)～(6))。このとき、すべての制約違反条件を成立させないような変数値が存在しない場合には新たな制約違反条件を生成し(S8)、格納する(S10)。この処理で、制約違反条件をすべて成立させないように開始時刻を設定でき、新たなスケジュールを得る(図15のスケジュール)。

【0074】このスケジュールでは締め切り時刻 $E=40$ なので、次に $E=39$ としてスケジューリング処理を繰り返す(図11〔2〕(7)～(14))。まず、成立してしまう制約違反条件を探し(S3)、その制約違反条件に含まれる変数値を変更し(S4)、その他の変数値を決定する(S7)。いずれかの制約違反条件が成立する場合には、新たな制約違反条件を生成し(S8)、S4に戻りスケジューリングを繰り返す。

【0075】以上の処理(図11〔2〕(7)～(14))で、すべての手順の開始時刻が決定でき、新たなスケジュールが作成される(図16のスケジュール)。このスケジュールの締め切り時刻 $E=34$ であり、次に $E=33$ としてスケジューリング処理を繰り返す(図12〔2〕(15)～(18))。そして、また、新たなスケジュールが得られる(図17のスケジュール)。

【0076】このスケジュールの締め切り時刻 $E=32$ なので、次に $E=31$ としてスケジューリング処理を繰り返す(図12〔2〕(19)～(21)及び図13〔2〕(22)～(29))。処理をしていくと、新たな制約違反条件を生成した際、新たな制約違反条件の左辺に変数がない状態が起こる(図13〔2〕(29)の $32>E$)。これによってS9のNとなりスケジューリングは失敗する。

【0077】以上の処理により、スケジューリングが失敗する前に得られたスケジュール(図17のスケジュール)が最適スケジュールとなる。総所要時間は32である。

【0078】

【発明の効果】本発明によれば、一つの手順の開始時刻に対応する変数が数値化された制約違反条件を生成することにより、開始時刻を変更すべき手順を絞り込むとともに、その開始時刻の範囲を絞り込むことが可能にな

り、さらに、複数の不等式の論理積を取ることで、複数の仕事の絡みにより生じる制約違反のための十分条件が表現可能となる。この結果、無駄な探索が省略でき、効率的なスケジューリングを実行することが可能となる。

【0079】以上に述べた発明の効果は、制約充足条件を使用する場合にも同様に得られることは言うまでもない。

【図面の簡単な説明】

【図1】本発明の原理構成図（その1）である。
 【図2】本発明の原理構成図（その2）である。
 【図3】一実施例のシステム構成図である。
 【図4】一実施例の動作フローチャートである。
 【図5】一実施例の処理説明図（その1）である。
 【図6】一実施例の処理説明図（その2）である。
 【図7】一実施例の処理説明図（その3）である。
 【図8】一実施例のスケジューリング例（問題1について）である。
 【図9】4仕事3機械のジョブショップスケジューリング処理の説明図（その1）である。
 【図10】4仕事3機械のジョブショップスケジューリ 20
 ング処理の説明図（その2）である。
 【図11】4仕事3機械のジョブショップスケジューリ
 ング処理の説明図（その3）である。

【図12】4仕事3機械のジョブショップスケジューリ
 ング処理の説明図（その4）である。

【図13】4仕事3機械のジョブショップスケジューリ
 ング処理の説明図（その5）である。

【図14】4仕事3機械のジョブショップスケジューリ
 ング処理のスケジューリング例である（その1）。

【図15】4仕事3機械のジョブショップスケジューリ
 ング処理のスケジューリング例である（その2）。

10 【図16】4仕事3機械のジョブショップスケジューリ
 ング処理のスケジューリング例である（その3）。

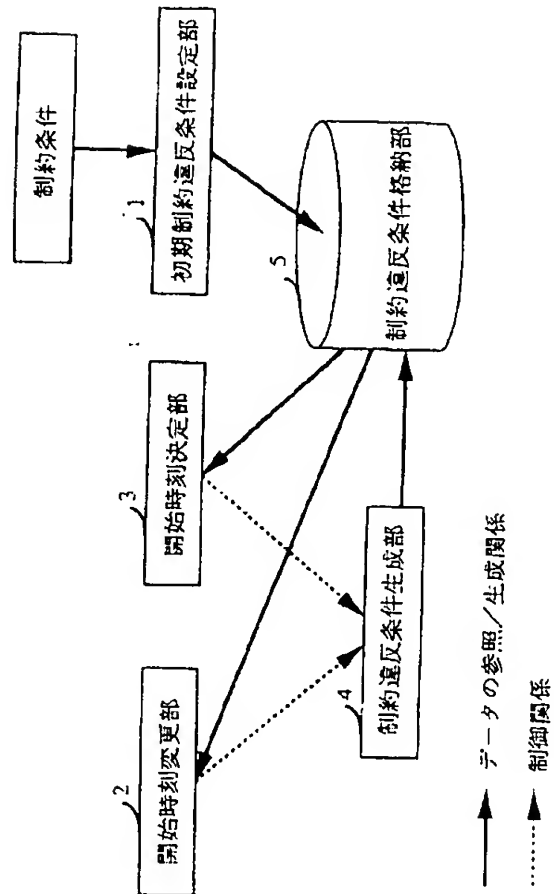
【図17】4仕事3機械のジョブショップスケジューリ
 ング処理のスケジューリング例である（その4）。

【符号の説明】

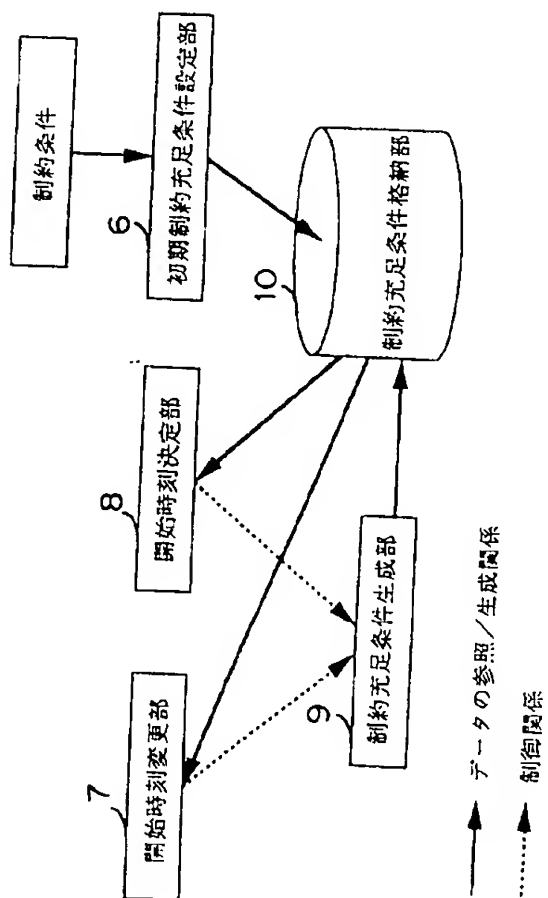
1 初期制約違反条件設定部
 2 開始時刻変更部
 3 開始時刻決定部
 4 制約違反条件生成部
 5 制約違反条件格納部
 6 初期制約充足条件設定部
 7 開始時刻変更部
 8 開始時刻決定部
 9 制約充足条件生成部
 10 制約充足条件格納部

【図1】

本発明の原理構成図(その1)

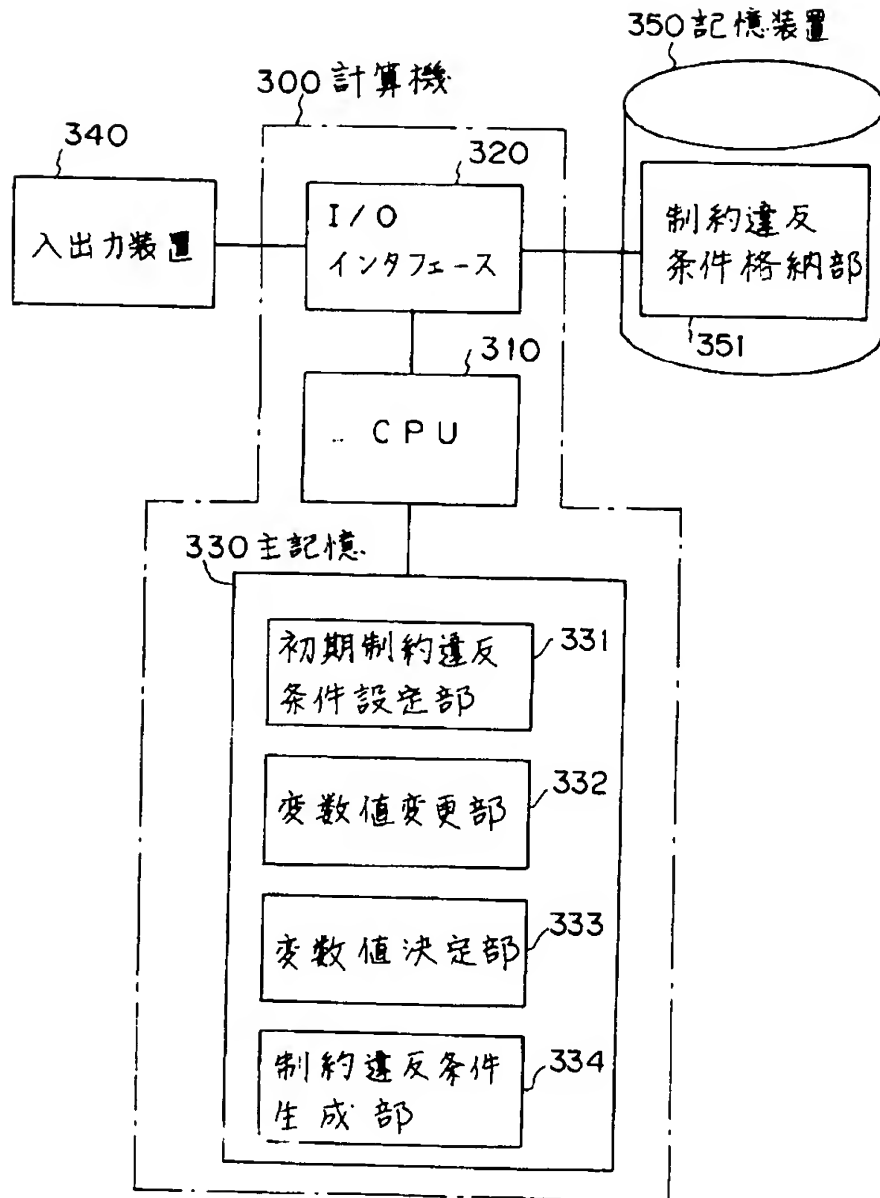


本発明の原理構成図(その2)



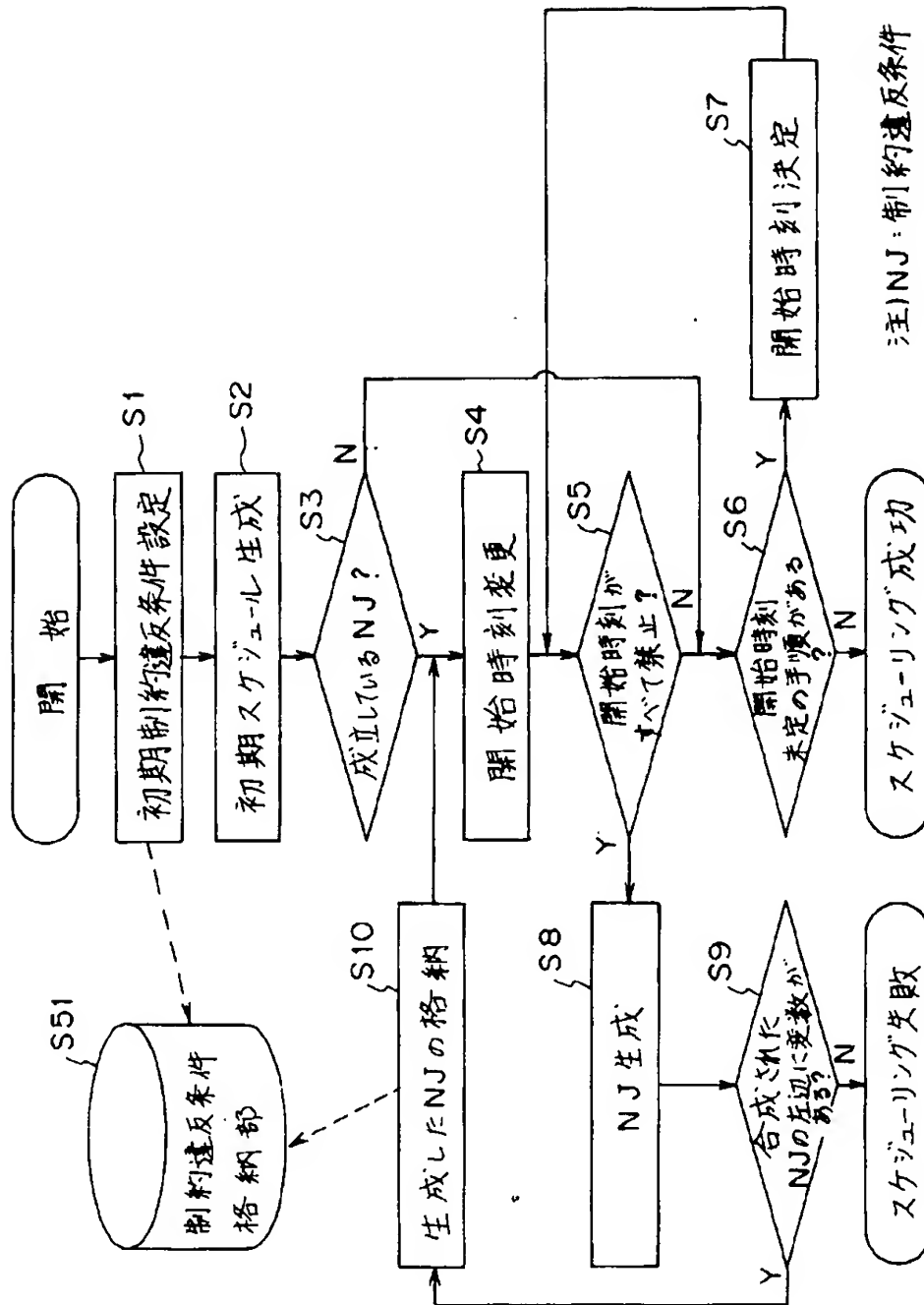
【図3】

一実施例のシステム構成図



【図4】

一実施例の動作フローチャート



【図5】

一実施例の処理説明図(その1)

〔問題1〕3仕事2機械のジョブショップスケジューリング問題で
最適スケジュール(総所要時間最小)を求める

仕事 \ 手順	1	2	仕事 \ 手順	1	2
1	1	2	1	6	4
2	2	1	2	4	3
3	1	2	3	2	5

使用機械 処理時間

〔1〕初期制約違反条件

(a)開始可能時刻に関する初期制約違反条件

$$\begin{array}{lll} x_{111} < 0 & x_{212} < 0 & x_{311} < 0 \\ x_{122} < 6 & x_{221} < 4 & x_{322} < 2 \end{array}$$

(b)締め切り時刻に関する初期制約違反条件

$$\begin{array}{lll} x_{111} + 10 > E & x_{212} + 7 > E & x_{311} + 7 > E \\ x_{122} + 4 > E & x_{221} + 3 > E & x_{322} + 5 > E \end{array}$$

(c)先行関係に関する初期制約違反条件

$$x_{111} + 6 > x_{122} \quad x_{212} + 4 > x_{221} \quad x_{311} + 2 > x_{322}$$

(d)複数手順の同時処理不可に関する初期制約違反条件

$$\begin{array}{l} x_{111} + 6 > x_{221} \wedge x_{221} + 3 > x_{111} \\ x_{111} + 6 > x_{311} \wedge x_{311} + 2 > x_{111} \\ x_{221} + 3 > x_{311} \wedge x_{311} + 2 > x_{221} \\ x_{122} + 4 > x_{212} \wedge x_{212} + 4 > x_{122} \\ x_{122} + 4 > x_{322} \wedge x_{322} + 5 > x_{122} \\ x_{212} + 4 > x_{322} \wedge x_{322} + 5 > x_{212} \end{array}$$

【図6】

一実施例の処理説明図(その2)

〔2〕スケジューリングのトレース

①の初期スケジュールから始める。

 $E=14$ とする。(1) $x_{12} + 5 > E$ が成立している。 $x_{12} = 0 \sim 1$ では $x_{12} < 2$ が成立。 $x_{12} = 2 \sim 7$ では $x_{11} + 2 > x_{12}$ が成立。これらの論理積は $x_{11} > 5$ 。 $x_{12} = 8 \sim 9$ では $x_{12} + 4 > x_{12} \wedge x_{12} + 5 > x_{12}$ が成立。これらの論理積は $x_{12} > 5 \wedge x_{12} < 13$ 。 $x_{12} = 10 \sim$ では $x_{12} + 5 > E$ が成立。これらの論理積は $15 > E$ 。従って、NJ は $15 > E \wedge x_{11} > 5 \wedge x_{12} > 5 \wedge x_{12} < 13$ 。(2) $x_{12} = 0 \sim 5$ では $x_{12} < 6$ が成立。 $x_{12} = 6 \sim 12$ では $15 > E \wedge x_{11} > 5 \wedge x_{12} > 5 \wedge x_{12} < 13$ が成立。これらの論理積は $15 > E \wedge x_{11} > 5$ 。 $x_{12} = 13 \sim$ では $x_{12} + 4 > E$ が成立。これらの論理積は $17 > E$ 。従って、NJ は $15 > E \wedge x_{11} > 5$ 。(3) $x_{11} = 0 \sim 5$ では $x_{11} + 6 > x_{11} \wedge x_{11} + 2 > x_{11}$ が成立。これらの論理積は $x_{11} > -1 \wedge x_{11} < 2$ 。 $x_{11} = 6 \sim$ では $15 > E \wedge x_{11} > 5$ が成立。これらの論理積は $15 > E$ 。従って、NJ は $15 > E \wedge x_{11} > -1 \wedge x_{11} < 2$ 。(4) $x_{11} = 2$ とする。(5) $x_{11} = 0$ とする。(6) $x_{12} = 8$ とする。(7) $x_{12} = 0 \sim 1$ では $x_{12} < 2$ が成立。 $x_{12} = 2 \sim 3$ では $x_{11} + 4 > x_{12} \wedge x_{12} + 5 > x_{12}$ が成立。これらの論理積は $x_{11} > -1 \wedge x_{12} < 7$ 。 $x_{12} = 4 \sim 11$ では $x_{12} + 4 > x_{12} \wedge x_{12} + 5 > x_{12}$ が成立。これらの論理積は $x_{12} > 7 \wedge x_{12} < 9$ 。 $x_{12} = 12 \sim$ では $x_{12} + 5 > E$ が成立。これらの論理積は $17 > E$ 。従って、NJ は $17 > E \wedge x_{11} > -1 \wedge x_{11} < 7 \wedge x_{12} > 7 \wedge x_{12} < 9$ 。

【図7】

一実施例の処理説明図(その3)

(8) $x_{12} = 9$ とする。

(9) $x_{32} = 1$ とする。

これで②のスケジューリングが得られた。

$E = 12$ とする。

(10) $x_{12} + 1 > E$ が成立している。

$x_{12} = 0 \sim 5$ では $x_{12} < 6$ が成立。

$x_{12} = 6 \sim 7$ では $x_{11} + 6 > x_{12}$ が成立。

これらの論理積は $x_{11} > 1$ 。

$x_{12} = 8$ では $17 > E \wedge x_{212} > -1 \wedge x_{212} < 7 \wedge x_{12} > 7 \wedge x_{12} < 9$ が成立。

$x_{12} = 9 \sim$ では $x_{12} + 1 > E$ が成立。

これらの論理積は $13 > E$ 。

従って、N J は $13 > E \wedge x_{11} > 1 \wedge x_{212} > -1 \wedge x_{212} < 7$ 。

(11) $x_{11} = 0 \sim 1$ では $15 > E \wedge x_{11} > -1 \wedge x_{11} < 2$ が成立。

これらの論理積は $15 > E$ 。

$x_{11} = 2 \sim$ では $13 > E \wedge x_{11} > 1 \wedge x_{212} > -1 \wedge x_{212} < 7$ が成立。

これらの論理積は $13 > E \wedge x_{212} > -1 \wedge x_{212} < 7$ 。

従って、N J は $13 > E \wedge x_{212} > -1 \wedge x_{212} < 7$ 。

(12) $x_{212} = 0 \sim 6$ では $13 > E \wedge x_{212} > -1 \wedge x_{212} < 7$ が成立。

これらの論理積は $13 > E$ 。

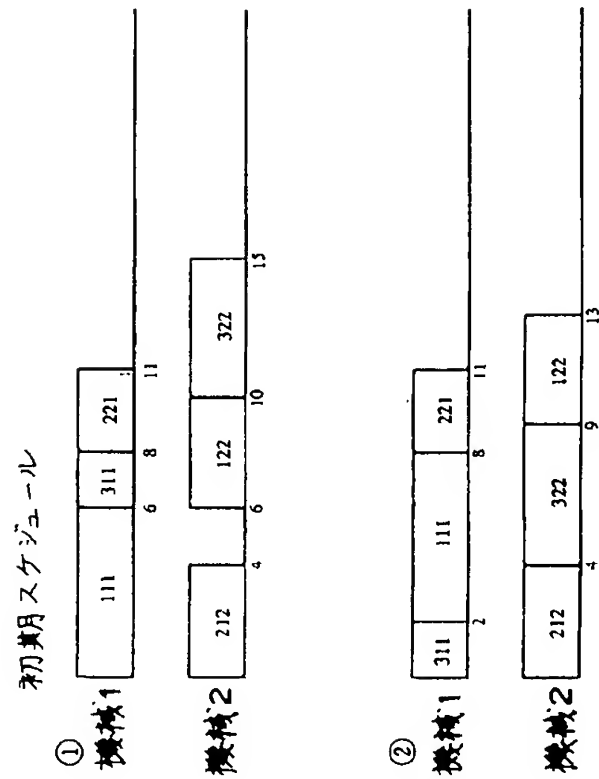
$x_{212} = 7 \sim$ では $x_{212} + 7 > E$ が成立。

これらの論理積は $14 > E$ 。

従って、最終的な N J は $13 > E$ 。②が最適スケジュールである。

【図8】

一実施例のスケジューリング例（問題1について）



【図9】

4仕事3手順のジョブショップスケジューリング処理の説明図(その1)

【問題2】 4仕事3機械のジョブショップスケジューリング問題について
最適スケジュールを求める

仕事 \ 手順	1	2	3
1	1	2	3
2	3	1	2
3	1	3	2
4	2	3	1

使用機械

仕事 \ 手順	1	2	3
1	5	8	2
2	7	3	9
3	1	7	10
4	4	11	7

処理時間

【1】初期制約違反条件

(a)開始可能時刻に関する初期制約違反条件

$$\begin{array}{llll}
 x_{111} < 0 & x_{213} < 0 & x_{311} < 0 & x_{412} < 0 \\
 x_{122} < 5 & x_{221} < 7 & x_{323} < 1 & x_{423} < 4 \\
 x_{133} < 13 & x_{232} < 10 & x_{332} < 8 & x_{431} < 15
 \end{array}$$

(b)締め切り時刻に関する初期制約違反条件

$$\begin{array}{llll}
 x_{111} + 15 > E & x_{213} + 19 > E & x_{311} + 18 > E & x_{412} + 22 > E \\
 x_{122} + 10 > E & x_{221} + 12 > E & x_{323} + 17 > E & x_{423} + 18 > E \\
 x_{133} + 2 > E & x_{232} + 9 > E & x_{332} + 10 > E & x_{431} + 7 > E
 \end{array}$$

(c)先行関係に関する初期制約違反条件

$$\begin{array}{llll}
 x_{111} + 5 > x_{122} & x_{122} + 8 > x_{133} & x_{213} + 7 > x_{221} & x_{221} + 3 > x_{232} \\
 x_{311} + 1 > x_{323} & x_{323} + 7 > x_{332} & x_{412} + 4 > x_{423} & x_{423} + 11 > x_{431}
 \end{array}$$

(d)複数手順の同時処理不可に関する初期制約違反条件

$$x_i + 1_i > x_j + 1_j > x_i \quad (\text{ただし、} i \text{ と } j \text{ は同一機械で処理される2つの手順})$$

【図10】

4 仕事3機械のジョブジョブスケジューリング処理の説明図(その2)

〔2〕スケジューリング・トレース

①の初期スケジューリングから始める。

E=42とする。

(1) $x_{32} + 17 > E$ が成立している ($x_{32} + 10 > E$ も成立しているが x_{32} を優先) $x_{32} = 0$ では $x_{32} < 1$ が成立。 $x_{32} = 1 \sim 5$ では $x_{31} + 1 > x_{32}$ が成立。これらの論理積は $x_{31} > 4$ 。 $x_{32} = 6$ では $x_{23} + 7 > x_{32} \wedge x_{32} + 7 > x_{23}$ が成立。 $x_{32} = 7 \sim 14$ では $x_{13} + 2 > x_{32} \wedge x_{32} + 7 > x_{13}$ が成立。これらの論理積は $x_{13} > 12 \wedge x_{13} < 14$ 。 $x_{32} = 15 \sim 23$ では $x_{43} + 11 > x_{32} \wedge x_{32} + 7 > x_{43}$ が成立。これらの論理積は $x_{43} > 14 \wedge x_{43} < 22$ 。 $x_{32} = 24 \sim$ では $x_{32} + 17 > E$ が成立。これらの論理積は $43 > E$ 。従って、N J は $43 > E \wedge x_{31} > 4 \wedge x_{23} > -1 \wedge x_{31} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14 \wedge x_{43} > 14 \wedge x_{43} < 22$ 。(2) $x_{43} = 0 \sim 3$ では $x_{43} < 4$ が成立。 $x_{43} = 4 \sim 6$ では $x_{23} + 7 > x_{43} \wedge x_{43} + 11 > x_{23}$ が成立。これらの論理積は $x_{23} > -1 \wedge x_{23} < 15$ 。 $x_{43} = 7 \sim 14$ では $x_{13} + 2 > x_{43} \wedge x_{43} + 11 > x_{13}$ が成立。これらの論理積は $x_{13} > 12 \wedge x_{13} < 18$ 。 $x_{43} = 15 \sim 21$ では $43 > E \wedge x_{31} > 4 \wedge x_{23} > -1 \wedge x_{23} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14 \wedge$ $x_{43} > 14 \wedge x_{43} < 22$ が成立。これらの論理積は $43 > E \wedge x_{31} > 4 \wedge x_{23} > -1 \wedge x_{23} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14$ 。 $x_{43} = 22 \sim 24$ では $x_{43} + 11 > x_{43}$ が成立。これらの論理積は $x_{43} < 33$ 。 $x_{43} = 25 \sim$ では $x_{43} + 18 > E$ が成立。これらの論理積は $43 > E$ 。従って、N J は $43 > E \wedge x_{31} > 4 \wedge x_{23} > -1 \wedge x_{23} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14 \wedge x_{43} < 33$ 。(3) $x_{43} = 33$ とする。(4) $x_{43} = 22$ とする。(5) $x_{31} = 15$ とする。(6) $x_{32} + 10 > E$ がまだ成立している。 $x_{31} = 22$ とする。

これで②のスケジューリングが得られた。

E=39とする。

【図11】

4 仕事3機械のジョブシッフ・スケジューリング処理の説明図(その3)

(7) $x_{42} + 18 > E$ が成立している ($x_{41} + 7 > E$ も成立しているが x_{42} を優先)

$x_{42} = 0 \sim 3$ では $x_{42} < 4$ が成立。

$x_{42} = 4 \sim 6$ では $x_{21} + 7 > x_{42} \wedge x_{42} + 11 > x_{21}$ が成立。

これらの論理積は $x_{21} > -1 \wedge x_{21} < 15$ 。

$x_{42} = 7 \sim 14$ では $x_{13} + 2 > x_{42} \wedge x_{42} + 11 > x_{13}$ が成立。

これらの論理積は $x_{13} > 12 \wedge x_{13} < 18$ 。

$x_{42} = 15 \sim 21$ では $43 > E \wedge x_{31} > 4 \wedge x_{21} > -1 \wedge x_{21} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14 \wedge x_{42} > 14 \wedge x_{42} < 22$ が成立。

これらの論理積は $43 > E \wedge x_{31} > 4 \wedge x_{21} > -1 \wedge x_{21} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14$ 。

$x_{42} = 22 \sim$ では $x_{42} + 18 > E$ が成立。

これらの論理積は $40 > E$ 。

従って、N J は $40 > E \wedge x_{31} > 4 \wedge x_{21} > -1 \wedge x_{21} < 13 \wedge x_{13} > 12 \wedge x_{13} < 14$ 。

(8) $x_{13} = 22$ とする。

(9) $x_{42} = 0 \sim 3$ では $x_{42} < 4$ が成立。

$x_{42} = 4 \sim 6$ では $x_{21} + 7 > x_{42} \wedge x_{42} + 11 > x_{21}$ が成立。

これらの論理積は $x_{21} > -1 \wedge x_{21} < 15$ 。

$x_{42} = 7 \sim 21$ では $x_{32} + 7 > x_{42} \wedge x_{42} + 11 > x_{32}$ が成立。

これらの論理積は $x_{32} > 14 \wedge x_{32} < 18$ 。

$x_{42} = 22 \sim$ では $x_{42} + 18 > E$ が成立。

これらの論理積は $40 > E$ 。

従って、N J は $40 > E \wedge x_{31} > -1 \wedge x_{21} < 15 \wedge x_{32} > 14 \wedge x_{32} < 18$ 。

(10) $x_{32} = 7$ とする。

(11) $x_{42} = 0 \sim 3$ では $x_{42} < 4$ が成立。

$x_{42} = 4 \sim 6$ では $x_{21} + 7 > x_{42} \wedge x_{42} + 11 > x_{21}$ が成立。

これらの論理積は $x_{21} > -1 \wedge x_{21} < 15$ 。

$x_{42} = 7 \sim 13$ では $x_{32} + 7 > x_{42} \wedge x_{42} + 11 > x_{32}$ が成立。

これらの論理積は $x_{32} > 6 \wedge x_{32} < 18$ 。

$x_{42} = 14 \sim 23$ では $x_{13} + 2 > x_{42} \wedge x_{42} + 11 > x_{13}$ が成立。

これらの論理積は $x_{13} > 21 \wedge x_{13} < 25$ 。

$x_{42} = 24 \sim$ では $x_{42} + 18 > E$ が成立。

これらの論理積は $42 > E$ 。

従って、N J は $42 > E \wedge x_{31} > -1 \wedge x_{21} < 15 \wedge x_{32} > 6 \wedge x_{32} < 18 \wedge x_{13} > 21 \wedge x_{13} < 25$ 。

(12) $x_{13} = 14$ とする。

(13) $x_{42} = 16$ とする。

(14) $x_{41} + 7 > E$ がまだ成立している。

$x_{41} = 27$ とする。

これで③のスケジューリングが得られた。

$E = 33$ とする。

【図12】

4仕事3機械のジョブショップスケジューリング処理の説明図(その4)

(15) $x_{41} + 18 > E$ が成立している ($x_{41} + 7 > E$ も成立しているが x_{42} を優先)

$x_{42} = 0 \sim 3$ では $x_{41} < 4$ が成立。

$x_{42} = 4 \sim 6$ では $x_{21} + 7 > x_{42} \wedge x_{42} + 11 > x_{31}$ が成立。

これらの論理積は $x_{21} > -1 \wedge x_{21} < 15$ 。

$x_{42} = 7 \sim 13$ では $x_{32} + 7 > x_{42} \wedge x_{42} + 11 > x_{32}$ が成立。

これらの論理積は $x_{32} > 6 \wedge x_{32} < 18$ 。

$x_{42} = 14 \sim 15$ では $x_{13} + 2 > x_{42} \wedge x_{42} + 11 > x_{13}$ が成立。

これらの論理積は $x_{13} > 13 \wedge x_{13} < 25$ 。

$x_{42} = 16 \sim$ では $x_{42} + 18 > E$ が成立。

これらの論理積は $34 > E$ 。

従って、 NJ は $34 > E \wedge x_{21} > -1 \wedge x_{21} < 15 \wedge x_{32} > 6 \wedge x_{32} < 18 \wedge x_{13} > 13 \wedge x_{13} < 25$ 。

(16) $x_{13} = 25$ とする。

(17) $x_{41} = 14$ とする。

(18) $x_{41} + 7 > E$ がまだ成立している。

$x_{42} = 25$ とする。

これで④のスケジューリングが得られた。

$E = 31$ とする。

(19) $x_{41} + 18 > E$ が成立している ($x_{41} + 7 > E$ も成立しているが x_{42} を優先)

$x_{42} = 0 \sim 3$ では $x_{41} < 4$ が成立。

$x_{42} = 4 \sim 6$ では $x_{31} + 7 > x_{42} \wedge x_{42} + 11 > x_{31}$ が成立。

これらの論理積は $x_{31} > -1 \wedge x_{31} < 15$ 。

$x_{42} = 7 \sim 13$ では $x_{32} + 7 > x_{42} \wedge x_{42} + 11 > x_{32}$ が成立。

これらの論理積は $x_{32} > 6 \wedge x_{32} < 18$ 。

$x_{42} = 14 \sim$ では $x_{41} + 18 > E$ が成立。

これらの論理積は $32 > E$ 。

従って、 NJ は $32 > E \wedge x_{31} > -1 \wedge x_{31} < 15 \wedge x_{32} > 6 \wedge x_{32} < 18$ 。

(20) $x_{32} = 0$ では $x_{32} < 1$ が成立。

$x_{42} = 1 \sim 5$ では $x_{31} + 1 > x_{42}$ が成立。

これらの論理積は $x_{31} > 4$ 。

$x_{42} = 6$ では $x_{23} + 7 > x_{42} \wedge x_{32} + 7 > x_{21}$ が成立。

$x_{42} = 7 \sim 17$ では $32 > E \wedge x_{31} > -1 \wedge x_{31} < 15 \wedge x_{32} > 6 \wedge x_{32} < 18$ が成立。

これらの論理積は $32 > E \wedge x_{21} > -1 \wedge x_{31} < 15$ 。

$x_{42} = 18 \sim$ では $x_{32} + 17 > E$ が成立。

これらの論理積は $35 > E$ 。

従って、 NJ は $32 > E \wedge x_{21} > -1 \wedge x_{31} < 15 \wedge x_{31} > 4$ 。

(21) $x_{31} = 0 \sim 4$ では $x_{31} + 5 > x_{31} \wedge x_{31} + 1 > x_{31}$ が成立。

これらの論理積は $x_{31} > -1 \wedge x_{31} < 1$ 。

$x_{31} = 5 \sim$ では $32 > E \wedge x_{21} > -1 \wedge x_{31} < 13 \wedge x_{31} > 4$ が成立。

これらの論理積は $32 > E \wedge x_{21} > -1 \wedge x_{31} < 13$ 。

従って、 NJ は $32 > E \wedge x_{21} > -1 \wedge x_{31} < 1 \wedge x_{31} > -1 \wedge x_{31} < 13$ 。

【図13】

4 仕事3機械のジョブジョッパスケジューリング処理の説明図(その5)

(22) $x_{111} = 1$ とする。

(23) $x_{111} = 0$ とする。

(24) $x_{121} = 0$ では $x_{121} < 1$ が成立。

$x_{121} = 1 \sim 6$ では $x_{211} + 7 > x_{121} \wedge x_{121} + 7 > x_{211}$ が成立。

これらの論理積は $x_{211} > -1 \wedge x_{211} < 8$ 。

$x_{121} = 7 \sim 17$ では $32 > E \wedge x_{211} > -1 \wedge x_{211} < 15 \wedge x_{121} > 6 \wedge x_{121} < 18$ が成立。

これらの論理積は $32 > E \wedge x_{211} > -1 \wedge x_{211} < 15$ 。

$x_{121} = 18 \sim$ では $x_{121} + 17 > E$ が成立。

これらの論理積は $35 > E$ 。

従って、N J は $32 > E \wedge x_{211} > -1 \wedge x_{211} < 8$ 。

(25) $x_{211} = 8$ とする。

(26) $x_{211} = 1$ とする。

(27) $x_{421} = 0 \sim 3$ では $x_{421} < 4$ が成立。

$x_{421} = 4 \sim 7$ では $x_{321} + 7 > x_{421} \wedge x_{421} + 11 > x_{321}$ が成立。

これらの論理積は $x_{321} > 0 \wedge x_{321} < 15$ 。

$x_{421} = 8 \sim 14$ では $x_{211} + 7 > x_{421} \wedge x_{421} + 11 > x_{211}$ が成立。

これらの論理積は $x_{211} > 7 \wedge x_{211} < 19$ 。

$x_{421} = 15 \sim$ では $x_{421} + 18 > E$ が成立。

これらの論理積は $33 > E$ 。

従って、N J は $33 > E \wedge x_{321} > 0 \wedge x_{321} < 15 \wedge x_{211} > 7 \wedge x_{211} < 19$ 。

(28) $x_{211} = 0 \sim 7$ では $32 > E \wedge x_{211} > -1 \wedge x_{211} < 8$ が成立。

これらの論理積は $32 > E$ 。

$x_{211} = 8 \sim 18$ では $33 > E \wedge x_{321} > 0 \wedge x_{321} < 15 \wedge x_{211} > 7 \wedge x_{211} < 19$ が成立。

これらの論理積は $33 > E \wedge x_{321} > 0 \wedge x_{321} < 15$ 。

$x_{211} = 19 \sim$ では $x_{211} + 19 > E$ が成立。

これらの論理積は $38 > E$ 。

従って、N J は $32 > E \wedge x_{321} > 0 \wedge x_{321} < 15$ 。

(29) $x_{321} = 0$ では $x_{321} < 1$ が成立。

$x_{321} = 1 \sim 14$ では $32 > E \wedge x_{321} > 0 \wedge x_{321} < 15$ が成立。

これらの論理積は $32 > E$ 。

$x_{321} = 15 \sim$ では $x_{321} + 17 > E$ が成立。

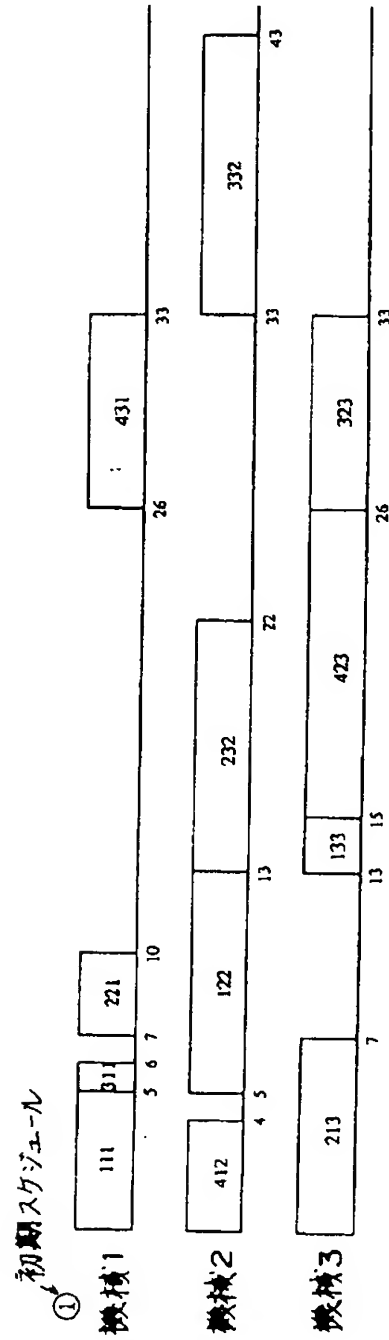
これらの論理積は $32 > E$ 。

従って、最終的な N J は $32 > E$ 。

【図14】

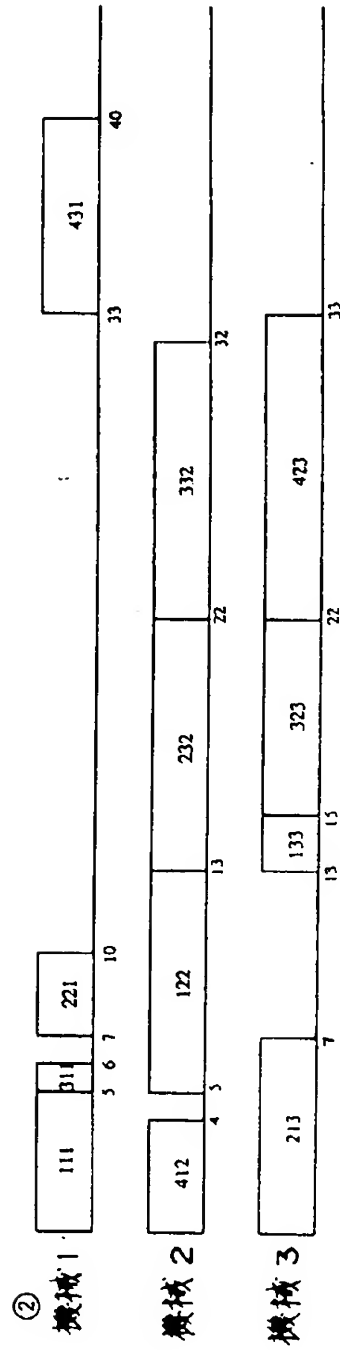
4仕事3機械のジョブショップスケジューリング問題のスケジュール例

(その1)



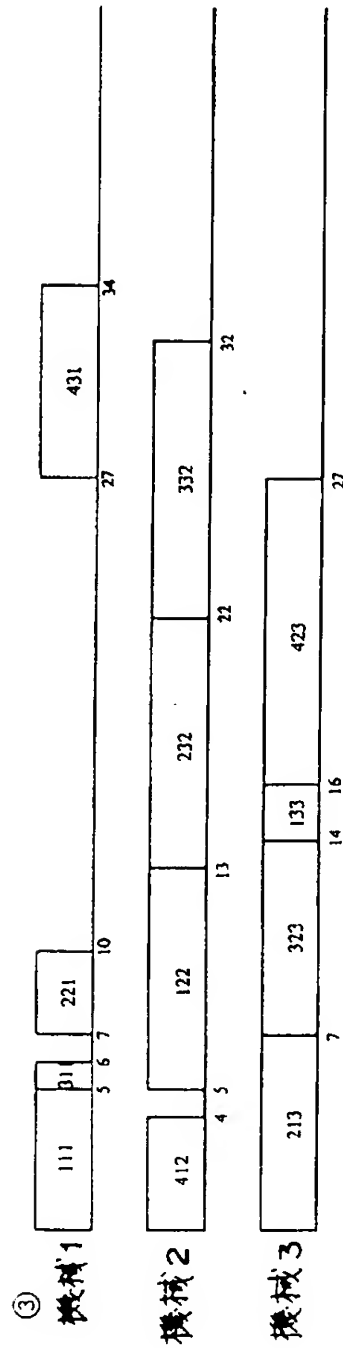
【図15】

4仕事3機械のジョブショップスケジューリング問題のスケジュール例
(その2)



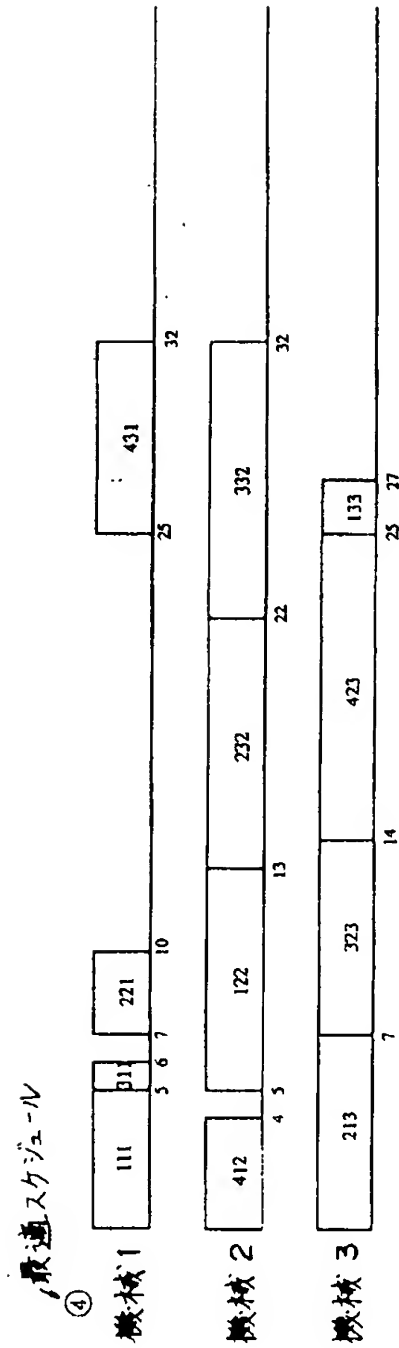
【図16】

4 仕事3機械のジョブショップスケジューリング問題のスケジュール例
(その3)



【図17】

4仕事3機械のジョブショップスケジューリング問題のスケジュール例
(その4)



フロントページの続き

(72)発明者 滝沢 ユカ
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内